project 3

Piere 2021-11-28

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 -
## v ggplot2 3.4.0      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() -
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Exercise 1

The database represents a digital media store. It contains information on tracks, artists, composers and a lot more. There is a specification on the database and its tables in the file hw_data/chinookDB.pdf. In this pdf you can see how the tables relate to each other and what columns exist. All questions should be answered with SQL code, except for those who asks for plots. If we ask for a mean, use SQL to compute it. We are going to do the following tasks:

a)We are going to extract all data from the tracks table using dbReadTable and plot the UnitPrice in a histogram. Then we will add the mean as a vertical line. Comment on your result.

b)Which genre has the least amount of tracks?

c)Which genre has the most amount of tracks in a playlist?

d)Which Composer (which is not NA) has most tracks in a playlist and what is that playlists name?

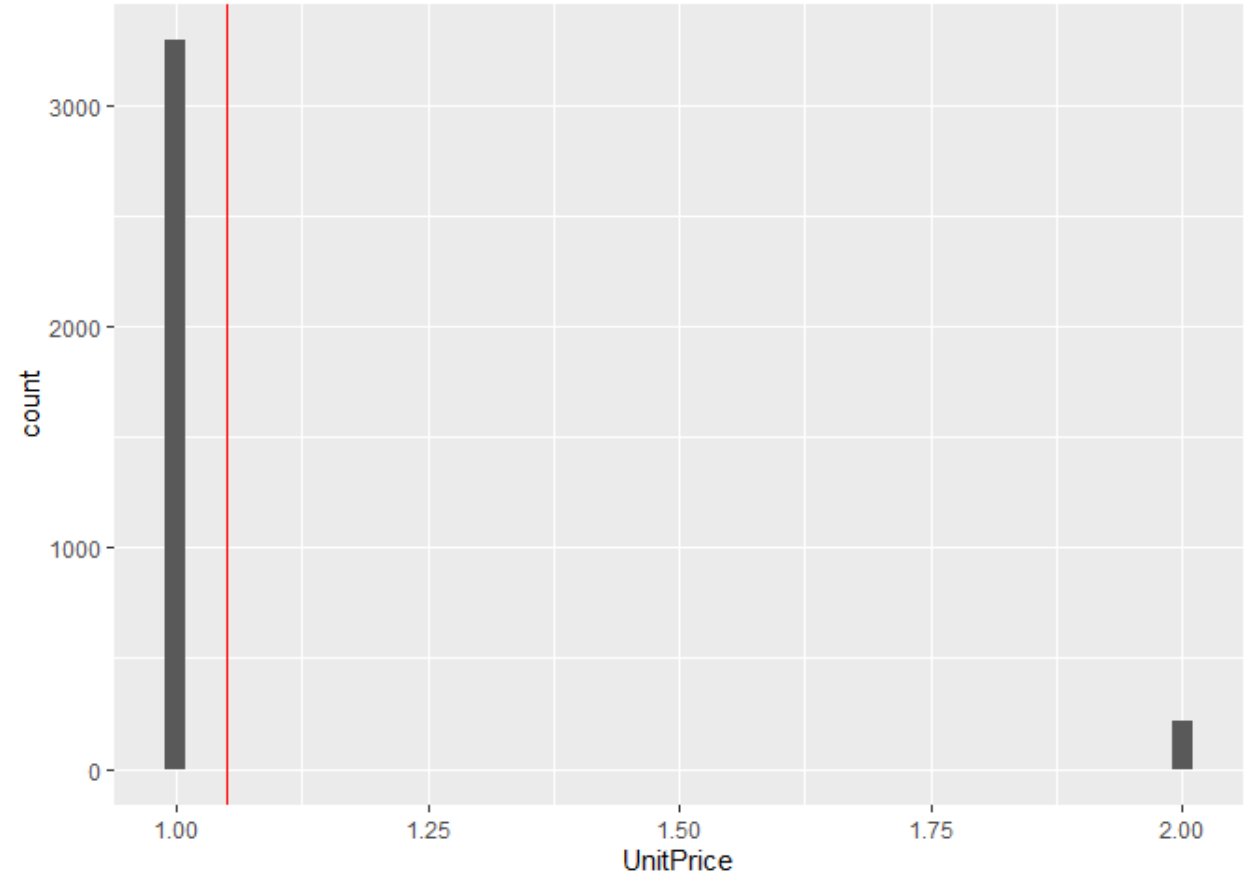
1a)

```
con <- DBI::dbConnect(RSQLite::SQLite(), "chinook.db") #makes a connection to the
con
```

```
## <SQLiteConnection>
##   Path: C:\Users\piere\OneDrive\Documents\MT4007 -data behandling\Homework\HW4\
```

```
##   Extensions: TRUE
```

```
tracks <- dbReadTable(con, "tracks")
#tracks

data1 <- dbReadTable(con, "tracks")%>%#exporting all tracks data
  ggplot(aes(x = UnitPrice)) + geom_histogram(bins = 50) + geom_vline(xintercept =
data1
```



Giving where the vertical red line is located on the x axis we can see that the mean is a bit above one dollar.

1b)

```
table_names <- dbListTables(con) #shows the names of the tables
#table_names

genre_count <- dbGetQuery(con, "select genres.Name, count(*) as 'number of tracks'
                          from tracks
                          join genres
                          on tracks.GenreId = genres.GenreId
                          group by tracks.GenreId
                          order by count(*)
```

```
                          limit(3)")
genre_count
```

```
##              Name number of tracks
## 1           Opera                1
## 2   Rock And Roll               12
## 3 Science Fiction               13
```

We can see that the least amount of tracks is the genre opera with only one single count.

1c)

```
playlist_track <- dbReadTable(con, "playlist_track")
#playlist_track

playlists <- dbReadTable(con, "playlists")
#playlists

genres <- dbReadTable(con, "genres")
#genres

tracks <- dbReadTable(con, "tracks")
#tracks

genre_playlist_tracks <- dbGetQuery(con, "select distinct playlists.Name as 'playl
                                    from tracks
                                    join genres
                                    join playlists
                                    join playlist_track
                                    on tracks.GenreId = genres.GenreId
                                    and playlist_track.PlaylistId = playlists.Play
                                    and playlist_track.TrackId = tracks.TrackId
                                    group by tracks.GenreId, playlist_track.Playli
                                    order by count(tracks.TrackId) desc
                                    limit(3)")
genre_playlist_tracks
```

```
##   playlist_name count(*) genre_name
## 1         Music     1297       Rock
## 2    90’s Music      621       Rock
## 3         Music      579      Latin
```

We can see that Rock has the most amount of track the the playlist Music.

1d)

```
#tracks
#playlist_track
#playlists


composer_most_tracks <- dbGetQuery(con, "select distinct Composer, count(*), playl
                                   from tracks
                                   join playlists
                                   join playlist_track
                                   on tracks.TrackId = playlist_track.TrackId
                                   and playlist_track.PlaylistId = playlists.Playl
                                   where Composer is not null
                                   group by Composer, playlist_track.PlaylistId
                                   order by count(tracks.TrackId) desc
                                   limit(3)")
composer_most_tracks
```

```
##          Composer count(*) playlist_name
## 1    Steve Harris       80         Music
## 2              U2       44         Music
## 3 Jagger/Richards       35         Music
```

We can see that the composer with the most tracks in a playlist is Steve Harris with a count of 80 tacks in the playlist Music.

Exercise 2

Data from this exercise originate from Skolverket and contain the grades of 6th graders from all elementary schools in Sweden. The data are freely available when aggregated at the municipality level and consists of a CSV file hw_data/exp_betyg_ak6_kommun_2018_19.csv containing the results of the year 2018/2019. Skolverket provides some additional information about the content of the data, if you click on “Analysstöd” at Skolverket’s data download page. Our tasks will be the following:

a)Make a plot which shows the difference of average grades between Boys and Girls in each county b)Make a map of Sweden where the municipalities are colored according to the event that the mean grade is higher in “Engelska” compared to “Idrott och hälsa” or not c)What conclusions can be drawn from looking at the map you produce? d)For each subject, compute the overall mean in Sweden. Do your conclusion in exercise c) still hold?

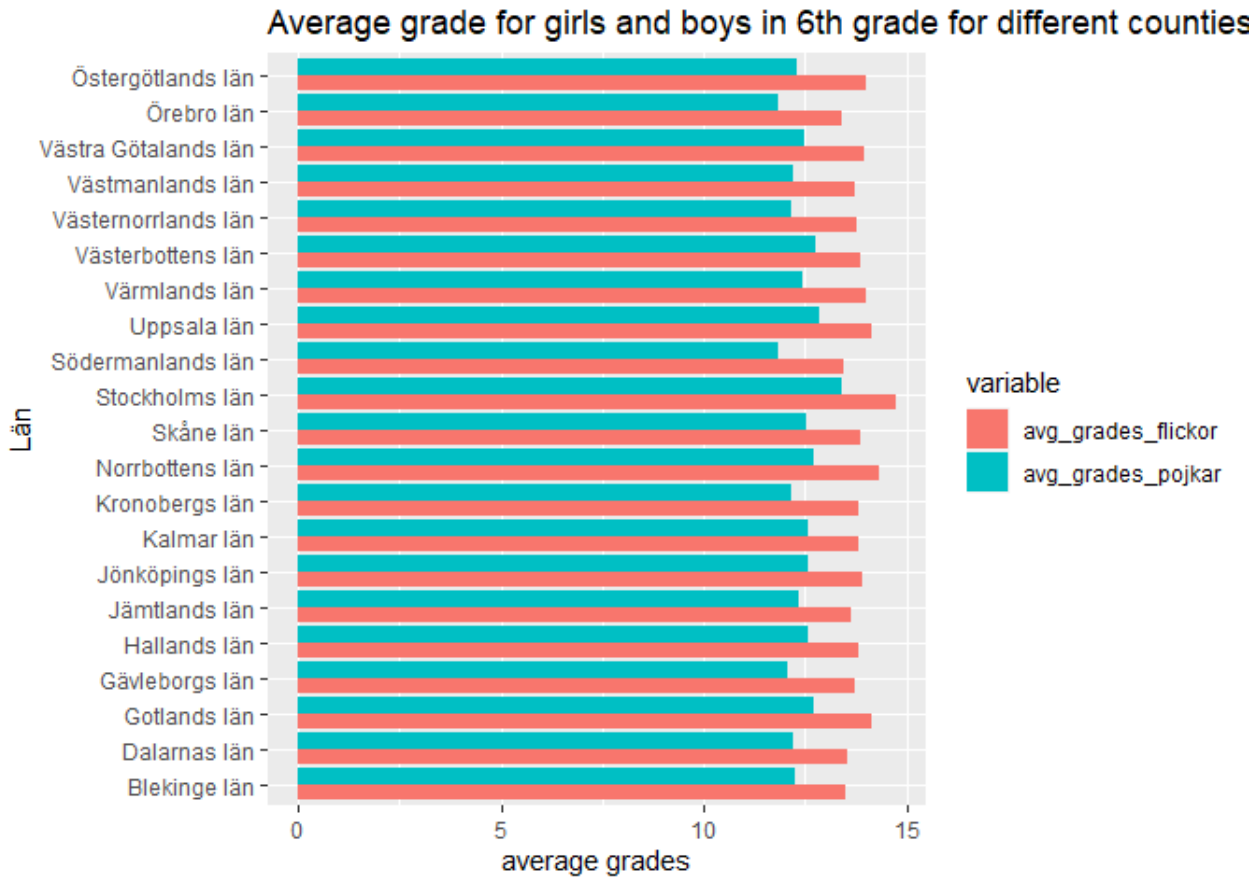
2a)

```
df <- read.csv("exp_betyg_ak6_kommun_2018_19.csv", sep = ";", header = T, skip = 6
  clean_names()%>%
   filter(typ_av_huvudman == "Samtliga") #restricting our attention to the result
#df
```

(-) means that the student never got a degree because of absence. If the result is based on less than 10 students it will get double doted (..). If the amount of student is between 1-4 that has not reached (A-E) then the portion of student that reached (A-E) will be showed as (~100). (.) means there is no information and the data is missing.

2b)

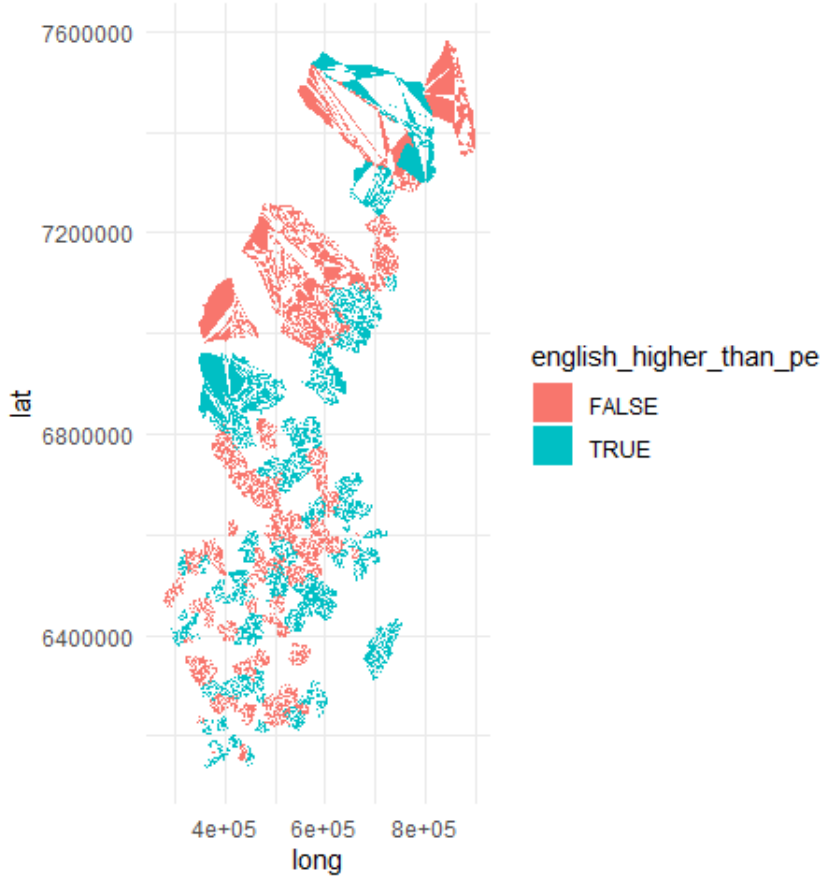
```
sixth_grades <- df%>%
  replace(is.na(.),0)%>% #replacing all NA values to 0
  mutate(flickor = as.numeric(gsub(" ", "", flickor)), flickor_2 = as.numeric(gsub
  mutate(total_grades_flickor = flickor * flickor_2, total_grades_pojkar = pojkar
  group_by(lan)%>%
  summarise(avg_grades_flickor = sum(total_grades_flickor)/sum(flickor), avg_grade

#sixth_grades

melt(sixth_grades, id = c("lan")) %>% #merging avg_grades_flickor/pojkar under the
 ggplot(aes(x=lan, y=value, fill = variable)) + geom_bar(stat="identity", position
```



Dealing with this problem was very challenging. First thing that was needed to do was to clean some variables for example changing the "," for a "." and then turning chr into numeric values. This game a problem which was some of the NA's did not really turned into 0 therefore it got difficult to perform any kind of computation. This problem was fixed by using gsub(). After that we could perform some operations which were first to multiply the amount of girls by their degree and then because of all the repetitive län (with different values) we needed to first sum all those and then divide by the total amount of girls in the län. Furthermore we used this information to create a new table which considered of three variables, avg_pojke/ flicka and län. This was also a problem because i had struggles ploting it so what i did was to merge both avg under a same column and by doing that it turned out to be easier to express the data as a bar plot which is shown above. The plot shows how the avg degree of girls is slightly bigger than the boy's. This kind of behavior could be explained by the amount of students, boys and girls.

2c) Make a map of Sweden where the municipalities are colored according to the event that the mean grade is higher in "Engelska" compared to "Idrott och hälsa" or not. In the csv file kommun-karta.csv we provide the borders of the municipalities in Sweden in polygon format. You can use the following example code as a basis for your solution.

```
municipalities <- read.csv("kommun_karTa.csv") %>%
  mutate(id = as.numeric(id))

#municipalities

##making sure to show the difference between english and pe. in other words we are

eng_idrott <- df %>%
  mutate(kommun_kod = as.numeric(kommun_kod))%>%
  filter(amne == c("Engelska", "Idrott och hälsa")) %>%
  mutate(totalt_2 = as.numeric(gsub(",","."", totalt_2)))%>%
  mutate(Idrott_och_hälsa = amne, Idrott_och_hälsa_betyg = totalt_2)%>%
  mutate_at(c("Idrott_och_hälsa", "Idrott_och_hälsa_betyg"), list(lead), n = 1)%>%
  filter(row_number() %% 2 == 1)%>%
  mutate(english_higher_than_pe = (totalt_2 > Idrott_och_hälsa_betyg))

#eng_idrott

merge(municipalities, eng_idrott, by.x = "id", by.y = "kommun_kod", all = FALSE)%>
  ggplot(aes(x = long, y = lat, group = id, fill = english_higher_than_pe)) + geom
```



Just a few comments over the plot. It does not look like it is filled so it forms the shape of Sweden but one can still see the shape of it and the municipalities where english score avg is higher than pe score avg. Looking at the graph we can somehow see that mostly of the map is covered by red which means false which means that physical education avg score is way more dominating than english avg score.

2d)

```
avg_grades <- df%>%
  replace(is.na(.),0)%>%
  select(totalt_2, amne)%>%
  mutate(totalt_2 = as.numeric(gsub(",",".",totalt_2)))%>%
  group_by(amne)%>%
  mutate(avg_all_grades = mean(totalt_2))%>%
  group_by(amne, avg_all_grades)%>%
  summarise()

avg_grades[c(3,8),]
```

```
## # A tibble: 2 x 2
## # Groups:   amne [2]
##   amne              avg_all_grades
##   <chr>                      <dbl>
```

```
## 1 Engelska                 13.4
## 2 Idrott och hälsa         13.6
```

We can see in the table that idrott och hälsa has a slightly higher avg score and english by 0.3 points. This concludes that the plot above really shows us that idrott has overall a higher avg score than english